REMARKS

Claims 1 – 19 and 28 – 31 are pending. Claims 20 – 27 have been cancelled without prejudice or disclaimer

Claims 12 and 15 have been amended for purposes of clarity.

New independent claim 28 and claims 29-31 depending therefrom have been added. Support can be found in the specification, including at paragraphs [0008] – [0010], [0026] – [0029], [0035] – [0036], and [0042].

I. 35 U.S.C. §102 Rejections

The existing rejections of claims 1 -27 as allegedly anticipated by U.S. Patent No. 6,178,432 (Cook et al.) were maintained in the Final Office Action mailed March 18, 2009.

As noted previously, anticipation under 35 U.S.C. § 102 requires that the applied reference teach the identical invention in which all of the claimed limitations are arranged or combined in the exact same way as recited in the claim. Assignee respectfully asserts that the Final Office Action has not established that Cook et al. anticipates the rejected or new claims and accordingly requests that the rejections be withdrawn.

A. Independent claims 1, 9, and 12 and Dependent Claims

In the reply dated December 3, 2008, Assignee's previous counsel argued that in Cook et al. the user navigates from visible to hidden objects that merely uncovers or hides existing or instantiated objects. In response, the March 18, 2009 Final Office Action alleges that the objects of Cook et al. are "not necessarily instantiated" based on portions at col. 15, lines 39-55, col. 10, lines 35-40, and Figure 7A of Cook et al.

Assignee respectfully notes that these portions are not directed to selective instantiation of objects. Col. 10, lines 35-40 merely notes that an applet must determine objects that contain behaviors triggered by an event—if an event is a mouse event the applet will look up all objects that contain the mouse position.

Col. 15, lines 39-55 discuss Figure 7A and inclusion of each object and structure listed in a prototype in a web page when the prototype is selected by the user. Additionally, at line 44, Cook et al. notes that "[s]ome prototypes contain a placeholder object which is bound to a user specified object when the user attaches the prototype to the user specified object. The user specified object to which the prototype is attached is referred to as the 'bound' object."

However, there is no indication as to how this pertains to instantiation of objects when an application is executed. Instead, the binding language appears to be related to design of prototypes and not runtime behavior. For instance, at col. 6, lines 15-30, Cook et al. places binding in context:

> Web page authoring module 208 is the product which enables a user to select and manipulate prototypes and to bind objects to those prototypes....By selecting and arranging prototypes and attaching the prototypes to objects, a user is able to specify an interactive Web page. Once the interactive web page is defined, the files associated with the page are stored on and distributed by Web server 210.

The remainder of col. 6 mentions use of a Java applet to manage display of objects and interaction of the user with objects and structures, but does not feature a teaching of instantiating some, but not all, objects.

Additionally, the Office Action relies on Figure 4B as teaching "generating a descriptor tree." However, although Figure 4B is a block diagram illustrating how a "tree" of triggered

events occurs, there is no indication as to how or why the tree-like shape of the diagram equates to a teaching of a descriptor tree as claimed.

Cook et al. mentions at col. 14, lines 15 that "a user may choose whether an object twill be downloaded separately from the rest of the objects on the interactive Web page only when activated by the user or whether the object will be loaded immediately when the page is accessed." However, this download functionality is not tied to the use of a descriptor tree, determination of visibility, or to instantiation.

Because Cook et al. has not been shown to teach each element of independent claims 1, 9, and 12, the rejections of those claims and the claims depending therefrom should be withdrawn.

B. New Independent claim 28 and Dependent Claims

New independent claim 28 is based on cancelled claim 20, but is not identical. Claim 28 is directed to a computer program product comprising:

- program code for accessing executable code of a rich internet application, the executable code comprising code for instantiating a plurality of objects, each object for rendering a corresponding interface element of the rich internet application;
- program code for identifying a subset of the plurality of objects, the subset comprising fewer than all of the plurality of objects;
- program code for instantiating the objects in the subset;
- program code for rendering an initial view of the application using the instantiated objects;

- program code for instantiating at least one other object of the plurality of objects in response to user interaction with an interface element of the initial view; and
- program code for rendering another view of the application using the instantiated at least one other object

As noted above, Cook et al. has not been shown to teach instantiating some, but not all (i.e., a subset) of the objects of a rich internet application. Nor has Cook et al. been shown to instantiate at least one other object in response to user interaction received after an initial view has been rendered. Thus, for at least these reasons, new independent claim 28 and claims 29-31 depending therefrom should be allowable over Cook et al.

III. Conclusion

For at least the reasons set forth above, Assignee respectfully requests that the rejections be withdrawn. No fee is believed due with this response. However, if a fee is due, please charge our Deposit Account No. 11-0855.

Respectfully submitted,

Date: 18 June 2009

/Eric G Zaiser/

Eric G. Zaiser
Reg. No. 58,352

KILPATRICK STOCKTON LLP
1100 Peachtree St. NE
Atlanta, GA 30309
404.745.2447 (voice)
404.541.3290 (fax)